IEEE CONTROL SYSTEMS LETTERS, VOL. 10, 2026

Composite Optimization in Open Multi-Agent Systems Under Additive Errors

Diego Deplano, *Member, IEEE*, Nicola Bastianello, *Member, IEEE*, Mauro Franceschelli, *Senior Member, IEEE*, and Karl H. Johansson, *Fellow, IEEE*

***Abstract*—This paper studies distributed composite optimization in open multi-agent systems, where agents may join and leave the network over time. We propose a variant of Open ADMM in which the composite proximal step is approximated through a finite number of proximal-gradient iterations, enabling agents to handle composite objectives even when the exact proximal operator is not available in closed form. Since the inexact computation of the proximal step induces additive errors, we establish a convergence result for paracontractive open multi-agent systems subject to bounded additive perturbations that capture both inexact local updates and noisy communications. Under mild assumptions on the network topology, the proposed algorithm converges linearly to a neighborhood of the optimal consensus set, with an explicit bound on the steady-state tracking error. The theoretical guarantees are corroborated by numerical experiments on distributed learning applications, namely linear regression (supervised learning) and principal component analysis (unsupervised learning).**

***Index Terms*—Robust optimization, distributed optimization, online optimization, online learning, classification, PCA, open networks.**

I. Introduction

DISTRIBUTED optimization over networks is a core primitive in large-scale learning and control, enabling agents to cooperatively estimate a common decision variable using only local objectives and peer-to-peer communication. In many learning and estimation tasks, the local costs naturally decompose into a *composite* form, consisting of a smooth loss term plus a (possibly non-smooth) regularizer or constraint, typically modeled by a *proximable* function [1], [2], [3], [4], [5], [6], [7], [8], [9]. This motivates the study of distributed *composite* optimization problems over networks of interacting agents, who seek agreement on a common model while leveraging local data and local structure.

The current literature on distributed composite optimization is limited to *closed* networks with a fixed set of agents, whereas practical deployments are often *open*: agents may join or leave due to mobility, availability, failures, privacy policies, economic incentives, and so on. Such scenarios fall under the umbrella of *open multi-agent systems*, where the number of participating agents is time-varying and classical convergence guarantees do not apply. In open networks, the consensus optimizer becomes inherently time-dependent, and join/leave events can prevent convergence, generate persistent transients, and even destabilize algorithms that are convergent in the static setting. Recent work has begun to formalize stability and performance notions tailored to open networks, and to develop analysis tools that expose the arising challenges [10], [11], [12], [13], while most contributions in distributed optimization for open networks remain algorithm- and application-specific; a broad account of these developments, which address non-composite optimization or consensus, is provided in [13].

The **main contribution** of this paper is to extend and analyze Open ADMM for distributed composite optimization problems over open networks. The specific novel contributions of the paper are as follows:

1) we specialize Open ADMM presented by us in [13] to deal with composite problems by approximating the local proximal update via a finite number of proximal-gradient inner iterations (Algorithm 1);
2) we characterize Open ADMM's convergence and performance by deriving an explicit bound that links tracking accuracy to the variability of the optimal solutions and to disagreement between local and global minimizers, and by showing linear convergence toward a neighborhood of the optimal consensus set (Theorem 1);
3) to prove the above, we establish a convergence result for paracontractive open multi-agent dynamics subject to bounded additive perturbations (Theorem 2), which constitutes a general analysis and design tool;
4) we showcase Open ADMM's performance on two representative learning tasks with local constraints: supervised linear regression and unsupervised distributed principal component analysis.

II. Preliminaries on Open Multi-Agent Systems

The sets of real and natural numbers are denoted by $\mathbb{R}$ and $\mathbb{N}$, respectively. Their positive subsets are denoted by $\mathbb{R}_+$ and

Received 17 March 2026; revised 13 May 2026; accepted 2 June 2026. Date of publication 9 June 2026; date of current version 19 June 2026. The work of Diego Deplano and Mauro Franceschelli was supported by the Fondazione di Sardegna with Project Formal Methods for Resiliency and Security in Networks of Systems (ReSNetS) call 2025 under Grant F83C26000350007. The work of Nicola Bastianello and Karl H. Johansson was supported in part by Swedish Research Council Distinguished Professor Grant 2017-01078 Knut and in part by the Alice Wallenberg Foundation Wallenberg Scholar Grant. This article has not been presented at a conference. Recommended by Senior Editor I.-C. Morarescu. *(Corresponding author: Diego Deplano.)*

Diego Deplano and Mauro Franceschelli are with DIEE, University of Cagliari, 09123 Cagliari, Italy (e-mail: diego.deplano@unica.it; mauro.franceschelli@unica.it).

Nicola Bastianello and Karl H. Johansson are with the School of Electrical Engineering and Computer Science, and Digital Futures, KTH Royal Institute of Technology, SE-100 44 Stockholm, Sweden (e-mail: nicolba@kth.se; kallej@kth.se).

Digital Object Identifier 10.1109/LCSYS.2026.3701867

$\mathbb{N}_+$, respectively. For examples and visual illustrations of the preliminaries reported next, we refer to [13, Sec. II].

Let $\mathcal{I} \neq \emptyset$ be a finite label set. Then, $x \in \mathbb{R}^{\mathcal{I}}$ denotes a point with labeled components x^i, $i \in \mathcal{I}$. To compare points in different labeled spaces, we use the concept of *open distance* from [13, Definition 3].

Definition 1 (Open Distance): The open distance between $x \in \mathbb{R}^{\mathcal{I}_1}$, $y \in \mathbb{R}^{\mathcal{I}_2}$, with $\mathcal{I}_1 \cap \mathcal{I}_2 \neq \emptyset$, is $d(x, y) = \|\tilde{x} - \tilde{y}\|_2$ where $\tilde{x} = [x^i \mid i \in \mathcal{I}_1 \cap \mathcal{I}_2]$ and $\tilde{y} = [y^i \mid i \in \mathcal{I}_1 \cap \mathcal{I}_2]$.

Given $x \in \mathbb{R}^{\mathcal{I}_0}$, $\mathcal{X} \subseteq \mathbb{R}^{\mathcal{I}_1}$, $\mathcal{Y} \subseteq \mathbb{R}^{\mathcal{I}_2}$, the distance from a point to a set and the distance between two sets, of possibly different labeled spaces, are defined by $d(x, \mathcal{Y}) := \inf_{y \in \mathcal{Y}} d(x, y)$ and $d(\mathcal{X}, \mathcal{Y}) := \inf_{x \in \mathcal{X}} d(x, \mathcal{Y})$. The set of points in $\mathcal{Y}$ at minimum distance from x is called the projection of x onto $\mathcal{Y}$ and is given by the *projection operator* $\text{proj}(x, \mathcal{Y}) = \{y^\star \in \mathcal{Y} : d(x, y^\star) = \inf_{y \in \mathcal{Y}} d(x, y)\}$. The projection operator enables the definition of the *shadow distance* between sets [13, Definition 4], capturing the worst-case distance between the corresponding projection points.

Definition 2 (Shadow Distance): The shadow distance between sets $\mathcal{X} \subseteq \mathbb{R}^{\mathcal{I}_1}$, $\mathcal{Y} \subseteq \mathbb{R}^{\mathcal{I}_2}$ with $\mathcal{I}_1 \cap \mathcal{I}_2 \neq \emptyset$ is

$$d_{\text{SH}}(\mathcal{X}, \mathcal{Y}) := \sup_{z \in \mathbb{R}^{\mathcal{I}_1 \cup \mathcal{I}_2}} d\big(\text{proj}(z, \mathcal{X}),\ \text{proj}(z, \mathcal{Y})\big).$$

The shadow distance yields the following triangle inequality:

$$d(z, \mathcal{X}) \leq d(z, \mathcal{Y}) + d_{\text{SH}}(\mathcal{X}, \mathcal{Y}). \tag{1}$$

The projection operator is a special case of the *proximal operator*, which, for a point $x \in \mathbb{R}^{\mathcal{I}_1}$, a function $f : \mathcal{Y} \to \mathbb{R}$ with $\mathcal{Y} \subseteq \mathbb{R}^{\mathcal{I}_2}$, and a penalty parameter $\varepsilon > 0$, is defined by

$$\text{prox}_f^{\varepsilon}(x) = \text{argmin}_{y \in \mathcal{Y}} \{f(y) + d^2(x, y)/2\varepsilon\}. \tag{2}$$

Indeed, $\text{proj}(x, \mathcal{Y}) = \text{prox}_{\iota_\mathcal{Y}}^{\varepsilon}(x)$ for $\iota_\mathcal{Y} : \mathbb{R}^{\mathcal{I}_2} \to \mathbb{R} \cup \{+\infty\}$ defined as $\iota_\mathcal{Y}(y) = 0$ if $y \in \mathcal{Y}$, and $\iota_\mathcal{Y}(y) = +\infty$ otherwise.

An *open multi-agent system* (OMAS) consists of a time-varying number $n_k \in \mathbb{N}_+$ of agents over time $k \in \mathbb{N}$. Interactions at time k are modeled by an undirected time-varying graph $\mathcal{G}_k = (\mathcal{V}_k, \mathcal{E}_k)$, where $\mathcal{V}_k$ is a finite node set with $|\mathcal{V}_k| = n_k$ and $\mathcal{E}_k \subseteq \mathcal{V}_k \times \mathcal{V}_k$ is the edge set. The neighbor set and degree of agent i are $\mathcal{N}_k^i := \{j \in \mathcal{V}_k \setminus \{i\} : (i, j) \in \mathcal{E}_k\}$ and $\eta_k^i := |\mathcal{N}_k^i|$. The agent set is partitioned into *remaining* $\mathcal{R}_k := \mathcal{V}_k \cap \mathcal{V}_{k-1}$, *arriving* $\mathcal{A}_k := \mathcal{V}_k \setminus \mathcal{V}_{k-1}$, and *departing* $\mathcal{D}_k := \mathcal{V}_k \setminus \mathcal{V}_{k+1}$ agents. Each agent $i \in \mathcal{V}_k$ has a state $x_k^i \in \mathbb{R}^{\mathcal{I}_k^i}$ of time-varying dimension $|\mathcal{I}_k^i|$, where $\mathcal{I}_k^i$ is the label set of its components. Without loss of generality, we let $\mathcal{I}_k^i = \mathcal{S}_k^i \times \mathcal{P}$, where $\mathcal{S}_k^i$ is a finite time-varying label set indexing the state blocks of agent i, and $\mathcal{P}$ is a fixed finite label set indexing the components of each block, so that $x_k^i = [\cdots, x_k^{ij}, \cdots]$ stacks vectors $x_k^{ij} \in \mathbb{R}^{\mathcal{P}}$ for all $i \in \mathcal{V}_k, j \in \mathcal{S}_k^i$. For each $i \in \mathcal{V}_k$, the set $\mathcal{S}_k^i$ is partitioned into *remaining* $\mathcal{R}_k^i := \mathcal{S}_k^i \cap \mathcal{S}_{k-1}^i$, *arriving* $\mathcal{A}_k^i := \mathcal{S}_k^i \setminus \mathcal{S}_{k-1}^i$, and *departing* $\mathcal{D}_k^i := \mathcal{S}_k^i \setminus \mathcal{S}_{k+1}^i$ labels of the i-th agent.

The label set indexing the network state $x_k \in \mathbb{R}^{\mathcal{I}_k}$ is $\mathcal{I}_k := \{(i, j, p) \mid i \in \mathcal{V}_k,\ j \in \mathcal{S}_k^i,\ p \in \mathcal{P}\}$, and $x_k = [\cdots; x_k^i; \cdots]$ stacks vectors $x_k^i \in \mathbb{R}^{\mathcal{S}_k^i \times \mathcal{P}}$ for all $i \in \mathcal{V}_k$. The evolution of each network state component $x_k^{ij} \in \mathbb{R}^{\mathcal{P}}$ combines a causal update for remaining labels of remaining agents with an initialization rule for arriving agents and arriving labels of remaining agents,

$$x_k^{ij} = \mathsf{T}_k^{ij}(x_{k-1}) := \begin{cases} \mathsf{F}_k^{ij}(x_{k-1}) & \text{if } i \in \mathcal{R}_k \wedge j \in \mathcal{R}_k^i, \\ x_k^{\text{A},ij} & \text{if } i \in \mathcal{A}_k \vee j \in \mathcal{A}_k^i, \end{cases} \tag{3}$$

which in compact form becomes

$$x_k = \mathsf{T}_k(x_{k-1}), \qquad x_k \in \mathbb{R}^{\mathcal{I}_k}. \tag{4}$$

If the agent and label sets are unchanged, then no initialization is needed and the state update rule T_k reduces to the map $\mathsf{F}_k : \mathbb{R}^{\mathcal{I}_{k-1}} \to \mathbb{R}^{\mathcal{I}_{k-1}}$, yielding the *standard iteration*

$$x_k = \mathsf{F}_k(x_{k-1}), \quad \text{when} \quad \mathcal{I}_k = \mathcal{I}_{k-1}. \tag{5}$$

In open systems, the concept of equilibrium point is replaced by that of a *point of interest*, i.e., a fixed point of the operator F_k in eq. (5). Under the assumption that at each time $k \in \mathbb{N}$ there is only one point of interest, the concept of a *trajectory of points of interest* was introduced in [10, Definition 3.1]; following [13, Definition 1], it can be generalized to the case of multiple points (sets) of interest.

Definition 3 (Trajectory of Sets of Interest): Consider an OMAS and assume that for each $k \in \mathbb{N}$ the equation $x = \mathsf{F}_{k+1}(x)$ admits at least one solution $\hat{x}_k \in \mathbb{R}^{\mathcal{I}_k}$, which is called point of interest, and let $\hat{\mathcal{X}}_k \subseteq \mathbb{R}^{\mathcal{I}_k}$ be the set of all points of interest at $k \in \mathbb{N}$, i.e.,

$$\hat{\mathcal{X}}_k := \{x \in \mathbb{R}^{\mathcal{I}_k} \mid x = \mathsf{F}_{k+1}(x)\}. \tag{6}$$

Then, the sequence $\{\hat{\mathcal{X}}_k : k \in \mathbb{N}\}$ is called the trajectory of sets of interest (TSI) of the OMAS.

The existence of a TSI is guaranteed for paracontractive OMASs [13, Definition 5].

Definition 4 (Paracontractivity): An OMAS is said to be paracontractive if there exists $\gamma \in (0, 1)$ such that for all $k \geq 0$ and for all $x \in \mathbb{R}^{\mathcal{I}_k}$ it holds

$$d(\mathsf{F}_{k+1}(x), \hat{\mathcal{X}}_k) \leq \gamma \cdot d(x, \hat{\mathcal{X}}_k), \tag{7}$$

where $\{\hat{\mathcal{X}}_k : k \in \mathbb{N}\}$ is the TSI of the OMAS.

Finally, stability is tailored to size-varying settings by referring it to the TSI rather than the set of equilibrium points. The following definitions are straightforward generalizations of [10, Definition 3.3] and [14, Definition 5] for OMASs with multiple points (sets) of interest at each time $k \in \mathbb{N}$.

Definition 5 (Open Stability): Consider an OMAS with state evolution $\{x_k : k \in \mathbb{N}\}$. Its TSI $\{\hat{\mathcal{X}}_k : k \in \mathbb{N}\}$ is said to be open stable if there is a stability radius $R \geq 0$ with the following property: for every $\varepsilon > R$, there is $\delta > 0$ such that:

$$\frac{d(x_0, \hat{\mathcal{X}}_0)}{\sqrt{|\mathcal{I}_0|}} < \delta \Rightarrow \frac{d(x_k, \hat{\mathcal{X}}_k)}{\sqrt{|\mathcal{I}_k|}} < \varepsilon, \quad \forall k \geq 0.$$

Definition 6 (Global Asymptotic Open Stability): Consider an OMAS whose TSI $\{\hat{\mathcal{X}}_k : k \in \mathbb{N}\}$ is open stable with stability radius $R \geq 0$. The TSI is said to be globally asymptotically open stable if all trajectories converge to within a distance of R from the TSI:

$$\limsup_{k \to \infty} \frac{d(x_k, \hat{\mathcal{X}}_k)}{\sqrt{|\mathcal{I}_k|}} \leq R.$$

III. Problem Formulation and Main Result

We consider the distributed optimization problem

$$\min_{y_i \in \mathbb{R}^{\mathcal{P}}} \sum_{i \in \mathcal{V}_k} \big(f_k^i(y_i) + g_k^i(y_i)\big) \ \text{ s.t. } y_i = y_j \ \forall (i, j) \in \mathcal{E}_k, \tag{8}$$

where $|\mathcal{P}|$ is the number of variables and $f_k^i + g_k^i : \mathbb{R}^{\mathcal{P}} \mapsto \mathbb{R}$ denotes the local, composite objective function of agent $i \in \mathcal{V}_k$ at time $k \in \mathbb{N}$. We aim to find an OMAS to solve (8) under suitable assumptions on the local objective functions and the corresponding local and global solutions. We formalize our set of assumptions in Assumption 1, which makes use of the following notation for the set of solutions to (8),

$$\mathcal{Y}_k^\star := \operatorname*{argmin}_{y\in\mathbb{R}^{\mathcal{P}}} \sum_{i\in\mathcal{V}_k} \left(f_k^i(y) + g_k^i(y)\right),$$

and for the minimizers of each local cost function,

$$\mathcal{Y}_k^{i,\star} = \operatorname*{argmin}_{y\in\mathbb{R}^{\mathcal{P}}} \left(f_k^i(y) + g_k^i(y)\right). \tag{9}$$

Assumption 1: Problem (8) is such that, $\forall k \in \mathbb{N}$: (i) f_k^i, g_k^i are proper, lower semi-continuous, and convex[1] for all $i \in \mathcal{V}_k$; (ii) f_k^i are uniformly L-smooth over $i \in \mathcal{V}_k$ and $k \in \mathbb{N}$, while functions g_k^i are proximable[2]; (iii) the set of minimizers $\mathcal{Y}_k^{i,\star} \subseteq \mathbb{R}^{\mathcal{P}}$ for each local composite cost function $f_k^i+g_k^i$ is nonempty; (iv) the distance between any two consecutive global solutions $y_k^\star \in \mathcal{Y}_k^\star$ and $y_{k-1}^\star \in \mathcal{Y}_{k-1}^\star$ is upper bounded by $\sigma \geq 0$; (v) the distance between any local solution $y_k^{i,\star} \in \mathcal{Y}_k^{i,\star}$ and any global solution $y_k^\star \in \mathcal{Y}_k^\star$ is upper bounded by $\omega \geq 0$.

Algorithm 1 Open ADMM (for Composite problems)

Input: Relaxation parameter $\alpha \in (0, 1)$, penalty $\rho > 0$, number of local iterations $\tau \in \mathbb{N}_+$, step-size $\pi \in (0, 2/(L + \rho\overline{\eta}))$
Output: Each agent returns y_k^i which is an approximate solution to the optimization problem in (8)
for $k = 0, 1, 2, \ldots$ **each agent** $i \in \mathcal{V}_k$**:**
  if $i \in \mathcal{A}_k$ **then** // arriving agent
    initializes the state to a local optimum

$$x_k^{ij} = \rho y_k^{i,\star}, \quad y_{k-1}^i = y_k^{i,\star}, \qquad y_k^{i,\star} \in \mathcal{Y}_k^{i,\star}, \forall j \in \mathcal{N}_k^i$$

  else if $i \in \mathcal{R}_k$ **then** // remaining agent
    receives y_{k-1}^j, x_{k-1}^{ji} from each neighbor $j \in \mathcal{R}_k^i$
    updates the remaining state variable

$$x_k^{ij} = (1-\alpha)x_{k-1}^{ij} - \alpha x_{k-1}^{ji} + 2\rho\alpha y_{k-1}^j, \quad \forall j \in \mathcal{R}_k^i \tag{11}$$

    initializes the new states to a local optimum

$$x_k^{ij} = \rho y_k^{i,\star} \text{ (see (9))}, \quad \forall j \in \mathcal{A}_k^i$$

  end if
  sets $r_0^i = y_{k-1}^i$ and approximates the proximal in (10) by:
  for $t = 0, 1, \cdots, \tau - 1$ **do**

$$r_{t+1}^i = \operatorname{prox}_{g_k^i}^{\pi} \left(r_t^i - \pi\left(\nabla f_k^i(r_t^i) + \rho\eta_k^i(r_t^i - \overline{x}_k^i)\right)\right) \tag{12}$$

  end for
  transmits $y_k^i = r_\tau^i$, x_k^{ij} to each neighbor $j \in \mathcal{N}_k^i$
end for

We detail in Algorithm 1 how to apply Open ADMM, presented by us in [13], to solve the composite problem (8). At each time $k \in \mathbb{N}$, each agent $i \in \mathcal{V}_k$ maintains and updates one edge variable $x_k^{ij} \in \mathbb{R}^{\mathcal{P}}$ for each neighbor $j \in \mathcal{N}_k^i$, using the information received through local communication as in (11). These variables are then aggregated into the weighted average $\overline{x}_k^i \in \mathbb{R}^{\mathcal{P}}$, which acts as the reference point of the local proximal step used to compute the estimate $y_k^i \in \mathbb{R}^{\mathcal{P}}$ of the global solution, namely,

$$y_k^i = \operatorname{prox}_{f_k^i+g_k^i}^{1/\rho\eta_k^i}\left(\overline{x}_k^i\right), \quad \overline{x}_k^i = \frac{1}{\rho\eta_k^i}\sum_{j\in\mathcal{N}_k^i} x_k^{ij}. \tag{10}$$

However, the local proximal operator in (10) may not admit a closed form, since only g_k^i is guaranteed to be proximable. We therefore exploit the composite structure of $f_k^i + g_k^i$ to approximate (10) through a finite number $\tau \in \mathbb{N}_+$ of *proximal gradient descent* iterations with step-size $\pi \in \mathbb{R}_+$ as in eq. (12) [15, Sec. 28.5]. Each agent $i \in \mathcal{V}_k$ has $\mathcal{S}_k^i = \mathcal{N}_k^i$ and the time-varying set of labels $\mathcal{I}_k^i = \mathcal{S}_k^i \times \mathcal{P}$. The operator $\mathsf{T} : \mathbb{R}^{\mathcal{I}_k} \to \mathbb{R}^{\mathcal{I}_k}$ with $\mathcal{I}_k = \{(i, j, p) \mid i \in \mathcal{V}_k, j \in \mathcal{N}_k^i, p \in \mathcal{P}\}$ describing Algorithm 1 thus can be formalized block-wise as

$$x_k^{ij} = \mathsf{T}_k^{ij}(x_{k-1}) = \begin{cases} \mathsf{F}_k^{ij}(x_{k-1}) + e_k^{ij} & \text{if } i \in \mathcal{R}_k \wedge \ j \in \mathcal{R}_k^i, \\ \rho y_k^{i,\star} & \text{if } i \in \mathcal{A}_k \vee j \in \mathcal{A}_k^i, \end{cases}$$

where F_k^{ij} is the operator characterizing (10)–(11), and e_k^{ij} accounts for the inexact computation of the proximal (the discrepancy between (10) and (12)). We note that in practical scenarios, additional sources of errors can contribute to e_k^{ij}, *e.g.*, inexact inter-agent communications due to quantization/compression [17]. These non-idealities are accommodated by assuming that the aggregate perturbation due to inexact local update and noisy communications is bounded.

Assumption 2: At all times $k \in \mathbb{N}$, each node $i \in \mathcal{V}_k$ updates its local variable y_k^i with an additive error $u_k^i \in \mathbb{R}^{\mathcal{P}}$ bounded by $\|u_k^i\| \leq \nu_u$ and receives information from each neighbor $j \in \mathcal{R}_k^i$ with an additive noise $v_k^{ij} \in \mathbb{R}^{\mathcal{P}}$ bounded by $\|v_k^{ij}\| \leq \nu_v$. Thus, the perturbation $e_k^{ij} = \alpha(v_k^{ij} + 2\rho u_{k-1}^j)$ on the state variable x_k^{ij} is bounded by $\|e_k^{ij}\| \leq \nu_v + 2\rho\nu_u =: \nu_e < \infty$.

Remark 1: By the definition of proximal in (2), computing (10) requires solving $\min_x f_k^i(x) + g_k^i(x) + \frac{\rho\eta_k^i}{2}\|x - \overline{x}_k^i\|^2$. The sum of the first and last term is $(\rho\eta_k^i)$-strongly convex [15, Definition 10.7 and Proposition 10.8] and $(L + \rho\eta_k^i)$-smooth. Consequently, the proximal gradient iteration (12) used to approximate (10) is contractive [18, Theorem 3.1] and, in turn, r_t^i converges linearly to the exact proximal, i.e., $\left\|y_k^i - \operatorname{prox}_{f_k^i+g_k^i}^{1/\rho\eta_k^i}\left(\overline{x}_k^i\right)\right\| \leq O(\zeta^\tau)$ with $\zeta = \max\{|1 - \pi(L + \rho\overline{\eta})|, |1 - \pi\rho\min_i \eta_k^i|\} \in (0, 1)$. Therefore, for any given bound $\nu_e \in (0, \infty)$, there always exists a sufficiently large τ so that Assumption 2 is satisfied.

Remark 2: In principle, r_0^i in (12) can be initialized arbitrarily. Inspired by [19], we set $r_0^i = y_{k-1}^i$, which creates a feedback loop that improves the approximation.

Under Assumption 3, the following convergence result shows the global asymptotic open stability of the TSI of an OMAS executing Algorithm 1, and hence the convergence of the agents' estimates to consensus on the optimal solution set.

Assumption 3: At all times $k \in \mathbb{N}_+$, the graph $\mathcal{G}_k$ of the OMAS is undirected and connected and the neighborhood size of every agent is uniformly bounded by $\overline{\eta}$, i.e., $\eta_k^i \leq \overline{\eta}$, $\forall i \in \mathcal{V}_k$.

Theorem 1: Consider an OMAS executing Algorithm 1 to distributedly solve the optimization problem (8) under Assumptions 1–3. If the OMAS is paracontractive with $\gamma \in (0, 1)$ and the departure process is bounded with $\beta \in (\gamma, 1)$, then the TSI is globally asymptotically open stable with radius

$$R = (\rho(\sigma + \omega) + \nu_e)/(1 - \gamma/\beta),$$

[1] As in Definitions 1.4, 1.21, and 8.1 of [15], respectively.

[2] The functions f_k^i are uniformly L-smooth if they are continuously differentiable and if their gradients ∇f_k^i are L-Lipschitz continuous for all $i \in \mathcal{V}_k$ and $k \in \mathbb{N}$. If the agents have different and time-varying local smoothness constants L_k^i, then L must satisfy $L \geq \sup_{k\in\mathbb{N}, i\in\mathcal{V}_k} L_k^i$. A function g_k^i is proximable if it has a closed-form/computationally cheap proximal, cf. [16].

with linear rate $\gamma/\beta \in (0,1)$. The stacking $\widetilde{y}_k = [\widetilde{y}_k^1; \cdots; \widetilde{y}_k^{n_k}]$ of the agents' estimates $\widetilde{y}_k^i := \mathbb{1}_{\eta_k^i} \otimes y_k^i$ linearly converges to a consensus state on the optimal solutions $\mathcal{C}_k^\star := \{\mathbb{1}_{|\mathcal{E}_k|} \otimes y_k^\star \mid y_k^\star \in \mathcal{Y}_k^\star\}$ within radius $\Delta = (R + \nu_e)/\rho$.

Proof: The proof is given in Section III-B. ∎

Remark 3: Exploiting [13, Proposition 1], one may ensure paracontractivity of the OMAS in practice by showing that the standard iteration F_k associated with the executed algorithm is both averaged (see [17, Definition 2]) and metrically subregular (see [17, Definition 1]).

For Open ADMM as in Algorithm 1, the iteration F_k is averaged because it is induced by the relaxed Peaceman–Rachford splitting applied to the dual of problem (8) [20], [21]. Metric subregularity, instead, is guaranteed when F_k admits affine lower and upper bounds [17, Proposition 1]. This condition is satisfied for several classes of optimization problems, including those with linear and quadratic costs whose proximal operators are affine, such as the problems we consider in Sections IV-A–IV-B (cf. [17, Sec. IV]).

Remark 4: The requirement $\beta > \gamma$ means that departures must be sufficiently slow compared with the contraction rate. As β approaches γ, the radius bound becomes conservative; this is a limitation of the analysis and does not necessarily reflect the actual convergence neighborhood observed in practice. Future work will investigate other metrics to overcome this limitation, e.g., infinity norms as in [14].

A. Stability of OMAS Subject to Additive Error

Global asymptotic open stability is established for an OMAS with state updates affected by bounded additive perturbations. Sufficient conditions for stability in the sense of Definition 6 are obtained by imposing limits on the TSI variation and on agents' joining and leaving processes, defined next.

Definition 7 (Bounded TSI): The TSI $\{\hat{\mathcal{X}}_k : k \in \mathbb{N}\}$ of an OMAS is said to have bounded variation if

$$\exists B \geq 0 : \quad d_{\mathrm{SH}}(\hat{\mathcal{X}}_k, \hat{\mathcal{X}}_{k-1}) \leq B \sqrt{|\mathcal{I}_k \cap \mathcal{I}_{k-1}|}, \quad \forall k \in \mathbb{N}.$$

Definition 8 (Bounded Arrival Process): The arrival process of an OMAS with TSI $\{\hat{\mathcal{X}}_k : k \in \mathbb{N}\}$ is said to be bounded if, letting $x_k^{\mathrm{A}} \in \mathbb{R}^{\mathcal{I}_k \setminus \mathcal{I}_{k-1}}$ be the vector stacking the components of all arriving labels, it holds

$$\exists H \geq 0 : \quad d(x_k^{\mathrm{A}}, \hat{\mathcal{X}}_k) \leq H \sqrt{|\mathcal{I}_k \setminus \mathcal{I}_{k-1}|}, \quad \forall k \in \mathbb{N}.$$

Definition 9 (Bounded Departure Process): The departure process of an OMAS is said to be bounded if

$$\exists \beta \in (0,1) : \quad \sqrt{|\mathcal{I}_k|} \geq \beta \sqrt{|\mathcal{I}_{k-1}|}, \quad \forall k \in \mathbb{N}.$$

The main convergence result is given next.

Theorem 2: Given an OMAS, if:

(a) it is paracontractive with $\gamma \in (0,1)$;
(b) it admits a TSI with bounded variation $B \geq 0$;
(c) its arrival process is bounded with $H \geq 0$;
(d) its departure process is bounded with $\beta \in (\gamma, 1)$;
(e) the error is bounded by $\nu_e \geq 0$ as in Assumption 2;

then the TSI is globally asymptotically open stable with radius

$$R = B + H + \nu_e/(1 - \gamma/\beta). \tag{13}$$

Also, the convergence has linear rate $\theta = \gamma/\beta \in (0,1)$ with

$$d(x_k, \hat{\mathcal{X}}_k)/\sqrt{|\mathcal{I}_k|} \leq \theta^k d(x_0, \hat{\mathcal{X}}_0)/(\sqrt{|\mathcal{I}_0|}) + (1 - \theta^k)R.$$

Proof: Let us split the state $x_k = [x_k^{\mathrm{R}}; x_k^{\mathrm{A}}]$ into two vectors such that $x_k^{\mathrm{R}} \in \mathbb{R}^{\mathcal{I}_k \cap \mathcal{I}_{k-1}}$ is the vector of the remaining components and $x_k^{\mathrm{A}} \in \mathbb{R}^{\mathcal{I}_k \setminus \mathcal{I}_{k-1}}$ is the vector of the new components. For any two consecutive steps it holds:

$$\begin{aligned}
d(x_k, \hat{\mathcal{X}}_k) &= \sqrt{d^2(x_k^{\mathrm{R}}, \hat{\mathcal{X}}_k) + d^2(x_k^{\mathrm{A}}, \hat{\mathcal{X}}_k)} \\
&\leq \sqrt{d^2(x_k^{\mathrm{R}}, \hat{\mathcal{X}}_k)} + \sqrt{d^2(x_k^{\mathrm{A}}, \hat{\mathcal{X}}_k)} \\
&= d(x_k^{\mathrm{R}}, \hat{\mathcal{X}}_k) + d(x_k^{\mathrm{A}}, \hat{\mathcal{X}}_k) \overset{(i)}{\leq} d(x_k^{\mathrm{R}}, \hat{\mathcal{X}}_k) + \sqrt{|\mathcal{I}_k|} H \\
&\overset{(ii)}{\leq} d(x_k^{\mathrm{R}}, \hat{\mathcal{X}}_{k-1}) + d_{\mathrm{SH}}(\hat{\mathcal{X}}_k, \hat{\mathcal{X}}_{k-1}) + \sqrt{|\mathcal{I}_k|} H \\
&\overset{(iii)}{\leq} d(x_k^{\mathrm{R}}, \hat{\mathcal{X}}_{k-1}) + \sqrt{|\mathcal{I}_k|}(B + H) \\
&\overset{(iv)}{\leq} d(\mathsf{F}_k(x_{k-1}) + e_k, \hat{\mathcal{X}}_{k-1}) + \sqrt{|\mathcal{I}_k|}(B + H) \\
&\overset{(v)}{\leq} d(\mathsf{F}_k(x_{k-1}), \hat{\mathcal{X}}_{k-1}) + \|e_k\| + \sqrt{|\mathcal{I}_k|}(B + H) \\
&\overset{(vi)}{\leq} d(\mathsf{F}_k(x_{k-1}), \hat{\mathcal{X}}_{k-1}) + \sqrt{|\mathcal{I}_k|}(B + H + \nu_e) \\
&\overset{(vii)}{\leq} \gamma d(x_{k-1}, \hat{\mathcal{X}}_{k-1}) + \sqrt{|\mathcal{I}_k|}(B + H + \nu_e),
\end{aligned}$$

where (i) holds by assumption (c) and $|\mathcal{I}_k \setminus \mathcal{I}_{k-1}| \leq |\mathcal{I}_k|$; (ii) holds by triangle inequality in (1) for the shadow distance in Definition 2; (iii) holds by assumption (b) and $|\mathcal{I}_k \cap \mathcal{I}_{k-1}| \leq |\mathcal{I}_k|$; (iv) holds because the vector x_k^{R} is entirely contained in $\mathsf{F}_k(x_{k-1}) + e_k$ by definition, where $e_k = [\cdots, e_k^{ij}, \cdots]^\top$ denotes the stacking of all error vectors for $i \in \mathcal{V}_k$, $j \in \mathcal{N}_k^i$; (v) holds by $\widetilde{\mathsf{F}}_k^{ij}(x_{k-1}) = \mathsf{F}_k^{ij}(x_{k-1}) + e_k^{ij}$ and the triangle inequality; (vi) holds because the block components satisfy $\|e_k^{ij}\| \leq \nu_e$ by assumption (e) and the number of blocks is upper bounded by $|\mathcal{I}_k|$ as $e_k \in \mathbb{R}^{\mathcal{I}_k}$; (vii) holds by assumption (a). Thus,

$$\frac{d(x_k, \hat{\mathcal{X}}_k)}{\sqrt{|\mathcal{I}_k|}} \leq \left(\frac{\gamma}{\beta}\right)^k \frac{d(x_0, \hat{\mathcal{X}}_0)}{\sqrt{|\mathcal{I}_0|}} + (B + H + \nu_e) \sum_{i=0}^{k-1} \left(\frac{\gamma}{\beta}\right)^i.$$

The proof is concluded by noticing that $\limsup_{k\to\infty} d(x_k, \hat{\mathcal{X}}_k)/\sqrt{|\mathcal{I}_k|} \leq (B + H + \nu_e)/\left(1 - \frac{\gamma}{\beta}\right)$. ∎

B. Proof of Theorem 1

The first part of the proof consists in showing that conditions $(a) - (e)$ of Theorem 2 hold: (a) Paracontractivity of the operator ruling the standard dynamics holds by assumption with $\gamma \in (0,1)$; (b) Boundedness of the variation of the TSI holds by [13, Lemma 1] with $B = \rho\sigma$; (c) Boundedness of the arrival process holds by [13, Lemma 2] with $H = \rho\omega$; (d) Boundedness of the departure process holds by assumption with $\beta \in (\gamma, 1)$; (e) boundedness of the additive error holds by Assumption 2; then, substituting H and B into (13) yields the radius. The second part of the proof follows by: $d(\widetilde{y}_k, \mathcal{C}_k^\star) =$

$$\begin{aligned}
&= \sqrt{\textstyle\sum_{i\in\mathcal{V}_k, j\in\mathcal{N}_k^i} d^2(y_k^i, y_k^\star)} = \sqrt{\textstyle\sum_{i\in\mathcal{V}_k} \eta_k^i\, d^2(y_k^i, y_k^\star)} \\
&\overset{(a)}{\leq} \sqrt{\textstyle\sum_{i\in\mathcal{V}_k} \eta_k^i \left((\rho\sqrt{\eta_k^i})^{-1} d(x_k^i, \hat{\mathcal{X}}_k) + \frac{\nu_e}{\rho}\right)^2} \\
&\overset{(b)}{\leq} \sqrt{\textstyle\sum_{i\in\mathcal{V}_k} \rho^{-2} d^2(x_k^i, \hat{\mathcal{X}}_k)} + \sqrt{\textstyle\sum_{i\in\mathcal{V}_k} \rho^{-2} \eta_k^i \nu_e^2} \\
&\overset{(c)}{=} \rho^{-1}(d(x_k, \hat{\mathcal{X}}_k) + \sqrt{|\mathcal{E}_k|}\,\nu_e),
\end{aligned}$$

where (b) holds by Minkowski's inequality, (c) holds by $\sum_{i\in\mathcal{V}_k}\eta_k^i = |\mathcal{E}_k|$, and (a) holds by

$$\begin{aligned}
d(y_k^i, \mathcal{Y}_k^\star) &= \inf_{y\in\mathcal{Y}_k^\star}\left\|y_k^i - y\right\| \overset{(a)}{\leq} \left\|y_k^i - y_k^\star\right\| \\
&\overset{(b)}{\leq} \left\|\mathrm{prox}_{f_k^i+g_k^i}^{1/\rho\eta_k^i} \tfrac{1}{\rho\eta_k^i}A_{i,k}^\top x_k^i + u_k^i - \mathrm{prox}_{f_k^i+g_k^i}^{1/\rho\eta_k^i} \tfrac{1}{\rho\eta_k^i}A_{i,k}^\top \hat{x}_k^i\right\| \\
&\overset{(c)}{\leq} \left\|\tfrac{1}{\rho\eta_k^i}A_{i,k}^\top(x_k^i - \hat{x}_k^i)\right\| + \tfrac{\nu_e}{\rho} \overset{(d)}{\leq} \left\|\tfrac{1}{\rho\eta_k^i}A_{i,k}^\top\right\|\left\|x_k^i - \hat{x}_k^i\right\| + \tfrac{\nu_e}{\rho} \\
&\overset{(e)}{=} \left(\rho\sqrt{\eta_k^i}\right)^{-1} d(x_k^i, \hat{\mathcal{X}}_k) + \tfrac{\nu_e}{\rho},
\end{aligned}$$

where (a) holds since $y_k^\star \in \mathcal{Y}_k^\star$; (b) holds by eqs. (10), where $A_{i,k}^\top$ is the matrix computing the sums of neighbors' states and $u_k^i \in \mathbb{R}^{\mathcal{P}}$ is the error on the local variable as defined in Assumption 2; (c) follows by the non-expansiveness of the proximal operator and by Assumption 2 yielding $\|u_k^i\| \leq \nu_u \leq (2\rho)^{-1}\nu_e \leq \frac{\nu_e}{\rho}$; (d) holds by choosing $\hat{x}_k = \mathrm{arginf}_{y\in\hat{\mathcal{X}}_k}\|x_k - y\|$; (e) holds because matrix $\frac{1}{\eta_k^i}A_{i,k}^\top$ is column-stochastic and its rows sum up to $1/\eta_k^i$. This means that the linear convergence of $\widetilde{y}_k$ to a neighborhood of $\widetilde{\mathcal{C}}_k^\star$ is implied by that of x_k to a neighborhood of $\hat{\mathcal{X}}_k$, which follows from the first part of the proof, up to the convergence radius:

$$\limsup_{k\to\infty}\frac{d(\widetilde{y}_k, \widetilde{\mathcal{C}}_k^\star)}{\sqrt{|\mathcal{P}||\mathcal{E}_k|}} \leq \limsup_{k\to\infty}\frac{d(x_k, \hat{\mathcal{X}}_k) + \sqrt{|\mathcal{E}_k|}\nu_e}{\rho\sqrt{|\mathcal{P}||\mathcal{E}_k|}} \leq \frac{R+\nu_e}{\rho}.$$

IV. Application to Distributed Learning With Local Constraints

This section showcases the performance of Open ADMM applied to composite (un)supervised learning problems, which, in real scenarios, must be solved in open networks because the participating data holders (e.g., devices, sensors, or local processors) may join or leave the network over time due to intermittent availability, mobility, failures, or communication constraints. We focus on problems with local constraints[3]:

$$\min_{y_i\in\mathcal{Y}_k^i}\sum_{i\in\mathcal{V}_k} f_k^i(y_i) \quad \text{s.t.} \quad y_i = y_j \quad \forall (i,j)\in\mathcal{E}_k, \tag{14}$$

where $\mathcal{Y}_k^i \subset \mathbb{R}^{\mathcal{P}}$, with $i \in \mathcal{V}_k$, are nonempty, closed, convex sets such that $\bigcap_{i\in\mathcal{V}_k}\mathcal{Y}_k^i \neq \emptyset$ for all $k\in\mathbb{N}$. This problem is a special case of (8) with the indicator function $g_k^i = \iota_{\mathcal{Y}_k^i}$, such that $g_k^i(y) = 0$ if $y \in \mathcal{Y}_k^i$ and $g_k^i(y) = +\infty$ otherwise. The performance will be evaluated in terms of:

(i) optimality error $\mathrm{err}_k := d(\bar{y}_k, \mathcal{Y}_k^\star)$, $\bar{y}_k = \sum_{i\in\mathcal{V}_k} y_k^i/|\mathcal{V}_k|$;
(ii) constraint violation $\mathrm{cv}_k := d\left(\bar{y}_k, \bigcap_{i\in\mathcal{V}_k}\mathcal{Y}_k^i\right)$.

A. Supervised Learning: Linear Regression

We apply Open ADMM to solve distributed linear regression problems given next:

$$f_k^i(y) = \frac{1}{2}\left\|{A_k^i}^\top y - b_k^i\right\|^2, \quad \mathcal{Y}_k^i = [\underline{u}_k^i, \overline{u}_k^i], \quad \forall i\in\mathcal{V}_k,$$

where $A_k^i = [\cdots, a_{k,h}^i, \cdots] \in \mathbb{R}^{p\times m_k^i}$ is the matrix collecting all the feature vectors, $b_k^i = [\cdots, b_{k,h}^i, \cdots]^\top \in \mathbb{R}^{m_k^i}$ is the vector collecting the labels, and $\underline{u}_k^i \in [-20, -15]$, $\overline{u}_k^i \in [-10, -5]$ define the box constraints. The data are randomly generated using `scikit-learn`. This problem belongs to the class of *supervised learning*, where each agent's local dataset has time-varying cardinality $m_k^i \in \mathbb{N}_+$ and contains pairs $(a_{k,h}^i, b_{k,h}^i)$ where $a_{k,h}^i \in \mathbb{R}^p$ are $p \in \mathbb{N}_+$ features and $b_{k,h}^i \in \mathbb{R}$ is the associated label. The goal is to learn a model y that predicts the label $b_{k,h}^i$ from the sample $a_{k,h}^i$. The number of arriving and departing agents is drawn from independent Poisson distributions with the same mean.

[3] See code https://github.com/nicola-bastianello/open-admm

TABLE I
Asymptotic Values of Optimality Error and Constraint Violation for Different Values of τ

Open ADMM	τ	1	2	5	10	1000
standard	err	0.2371	0.2368	0.2367	0.2367	0.2367
	cv	0.2594	0.2585	0.2587	0.2587	0.2587
with max-consensus	err	0.0126	0.0125	0.0125	0.0125	0.0125
	cv	0.0498	0.0499	0.0499	0.0499	0.0499

We start by displaying in Table I the performance of Open ADMM with different values of the local number of iterations τ, in terms of asymptotic optimality error and constraint violation. Interestingly, as τ increases err and cv decrease, plateauing as $\tau \geq 5$. This is due to the fact that the main source of suboptimality when $\tau \geq 5$ are the variations in the network, rather than the inexact proximal computations.

While the results achieved by Open ADMM are already satisfying, the particular structure of the constraints inspired the design of a variant algorithm with better performance. Since the local constraints are box constraints $\mathcal{Y}_k := \bigcap_{i\in\mathcal{V}_k}\mathcal{Y}_k^i = [\max_{i\in\mathcal{V}_k}\underline{u}_k^i, \min_{i\in\mathcal{V}_k}\overline{u}_k^i] =: [\underline{u}_k, \overline{u}_k]$, they can be estimated through max-consensus protocols, such as OSTDMC [14] or Open ADMM itself [13, Sec. IV.A]. An improvement can thus obtained by preceding with a max-consensus step on $(\underline{u}_k^i, -\overline{u}_k^i)$ to obtain a tighter estimate $\hat{\mathcal{Y}}_k^i = [\hat{\underline{u}}_k^i, \hat{\overline{u}}_k^i]$ of $\mathcal{Y}_k$; in our simulations, this max-consensus step is implemented using Open ADMM as in [13, Sec. IV.A]. Therefore, during the local step (12) of Open ADMM, the agents apply projections onto $\hat{\mathcal{Y}}_k^i$ rather than $\mathcal{Y}_k^i$. Table I shows that this modified algorithm outperforms Open ADMM by one order of magnitude, both in terms of optimality error and constraint violation.

B. Unsupervised Learning: Distributed PCA

We apply Open ADMM to solve a distributed principal component analysis (PCA) problem given next:

$$f_k^i(Y) = -\mathrm{tr}(YC_{i,k}), \qquad \mathcal{Y}_k^i = \mathcal{F}_{p,q}, \tag{15}$$

where $Y \in \mathbb{R}^{p\times p}$ is the optimization variable, $C_{i,k} = \frac{1}{m_{i,k}}A_k^i{A_k^i}^\top \in \mathbb{R}^{p\times p}$ is the local correlation matrix, $A_k^i = [\cdots, a_{k,h}^i, \cdots] \in \mathbb{R}^{p\times m_k^i}$ is the matrix collecting all the feature vectors, $\mathcal{F}_{p,q} := \{M \in \mathbb{R}^{p\times p} \mid 0 \preceq M \preceq 1,\ \mathrm{tr}(M) = q\}$ is the *Fantope* set [22, Sec. 2.3.2], with $M \preceq a$ denoting a symmetric matrix with eigenvalues less or equal than $a\in\mathbb{R}$, and $\mathrm{tr}(M) = \sum_{\ell=1}^p M_{\ell\ell}$. This problem belongs to the class of *unsupervised learning*, where each agent holds a time-varying dataset of unlabeled feature vectors $a_{k,h}^i \in \mathbb{R}^p$, and the goal is to learn a projection matrix $U_{q,k} \in \mathbb{R}^{p\times q}$, with $q < p$, maximizing the variance of the projected data [23]; each agent constructs $U_{q,k}$ from the q leading eigenvectors of its local solution Y_i. Datasets of heterogeneous size $m_i \in [50, 200]$ are randomly generated with a latent $2D$ subspace of five clusters embedded in a $p = 50$ feature space (SNR= 25), and PCA is performed with $q = 2$. Figure 1 reports, on a logarithmic scale,

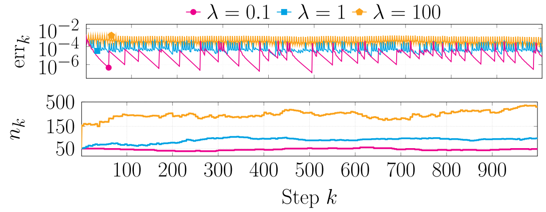

Fig. 1. Open ADMM applied to an unsupervised learning problem in an open network with $\lambda^{\text{join}} = \lambda^{\text{leave}} = \lambda \in \{0.1, 1, 100\}$.

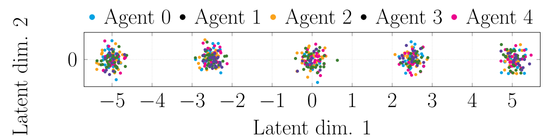

Fig. 2. Two-dimensional projection of the datasets for five randomly selected agents at a random time during the simulation when $\lambda = 100$.

the normalized tracking error err_k and the number of agents n_k. Simulations are run with join/leave events drawn from Poisson distributions with equal mean $\lambda^{\text{join}} = \lambda^{\text{leave}} = \lambda \in \{0.1, 1, 100\}$. Larger λ yields larger err_k due to stronger fluctuations in network size. After the initial transient, err_k settles within a bounded distance from the optimal solution, in agreement with the theoretical guarantees. Figure 2 shows a projection of the data at a random time in the worst-case $\lambda = 100$, confirming that the learned projection preserves the five-cluster structure.

V. Conclusion

This paper studied distributed *composite optimization* for OMAS by introducing a practical and effective variant of Open ADMM [13] in which local composite proximal steps are replaced by a finite number of proximal-gradient iterations. A reusable analysis tool for algorithm design in open networks is developed by establishing a general convergence result for paracontractive OMAS under bounded additive perturbations, which yields a linear tracking rate and a steady-state radius. Future work will focus on asynchronous communications, nonconvex learning settings, and robustness to attacks.

References

[1] N. K. Dhingra, S. Z. Khong, and M. R. Jovanovic, "The proximal augmented Lagrangian method for nonsmooth composite optimization," *IEEE Trans. Autom. Control*, vol. 64, no. 7, pp. 2861–2868, Jul. 2019. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85052638856doi=10.1109%2fTAC.2018.2867589partnerID=40md5=625d1586ac93f9d76684cb825729da31

[2] D. Yuan, Y. Hong, D. W. C. Ho, and S. Xu, "Distributed mirror descent for online composite optimization," *IEEE Trans. Autom. Control*, vol. 66, no. 2, pp. 714–729, Feb. 2021.

[3] R. Dixit, A. S. Bedi, and K. Rajawat, "Online learning over dynamic graphs via distributed proximal gradient algorithm," *IEEE Trans. Autom. Control*, vol. 66, no. 11, pp. 5065–5079, Nov. 2021.

[4] N. Bastianello and E. Dall'anese, "Distributed and inexact proximal gradient method for online convex optimization," in *Proc. Eur. Control Conf. (ECC)*, Jun. 2021, pp. 2432–2437.

[5] X. Li, L. Xie, and N. Li, "A survey on distributed online optimization and online games," *Annu. Rev. Control*, vol. 56, Aug. 2023, Art. no. 100904.

[6] R. Hou, Y. Yu, and X. Li, "Dynamic regret for distributed online composite optimization," *IEEE Trans. Autom. Control*, vol. 70, no. 5, pp. 3056–3071, May 2025.

[7] Z. Yu, D. W. C. Ho, D. Yuan, and Z. Shi, "Distributed composite optimization with distributed composite block mirror descent," *IEEE Trans. Autom. Control*, vol. 71, no. 6, pp. 4081–4088, Jun. 2026.

[8] H. Tian, X. Li, Z. Wu, and S. Magnússon, "Communication-efficient online federated composite optimization," *Automatica*, vol. 183, Jan. 2026, Art. no. 112679.

[9] Y. Zhou, L. Guo, X. Shi, and J. Cao, "A distributed proximal alternating direction multiplier method for multiblock nonsmooth composite optimization," *IEEE Trans. Control Netw. Syst.*, vol. 12, no. 1, pp. 202–215, Mar. 2025. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-105001067586doi=10.1109%2fTCNS.2024.3462519partnerID=40md5=3e53c986f0b1d96d1d55f49136ce7016

[10] M. Franceschelli and P. Frasca, "Stability of open multiagent systems and applications to dynamic consensus," *IEEE Trans. Autom. Control*, vol. 66, no. 5, pp. 2326–2331, May 2021.

[11] D. Deplano, M. Franceschelli, and A. Giua, "Stability of paracontractive open multi-agent systems," in *Proc. IEEE 63rd Conf. Decis. Control (CDC)*, Dec. 2024, pp. 3031–3036.

[12] R. Vizuete, C. M. de Galland, P. Frasca, E. Panteley, and J. M. Hendrickx, "Trends and questions in open multi-agent systems," in *Hybrid and Networked Dynamical Systems: Modeling, Analysis and Control*. Cham, Switzerland: Springer, 2024, pp. 219–252.

[13] D. Deplano, N. Bastianello, M. Franceschelli, and K. H. Johansson, "Optimization and learning in open multiagent systems," *IEEE Trans. Autom. Control*, vol. 71, no. 6, pp. 3864–3879, Jun. 2026.

[14] D. Deplano, M. Franceschelli, and A. Giua, "Distributed tracking of network size, diameter, radius, and node eccentricities in open multiagent systems," *IEEE Trans. Autom. Control*, vol. 71, no. 4, pp. 2274–2288, Apr. 2026.

[15] H. H. Bauschke and P. L. Combettes, *Convex Anal. Monotone Operator Theory Hilbert Spaces*. Cham, Switzerland: Springer, 2017.

[16] N. Parikh and S. Boyd, "Proximal algorithms," *Found. Trends Optim.*, vol. 1, no. 3, pp. 127–239, 2014.

[17] N. Bastianello, D. Deplano, M. Franceschelli, and K. H. Johansson, "Robust online learning over networks," *IEEE Trans. Autom. Control*, vol. 70, no. 2, pp. 933–946, Feb. 2025.

[18] A. B. Taylor, J. M. Hendrickx, and F. Glineur, "Exact worst-case convergence rates of the proximal gradient method for composite convex minimization," *J. Optim. Theory Appl.*, vol. 178, no. 2, pp. 455–476, Aug. 2018.

[19] X. Ren, N. Bastianello, K. H. Johansson, and T. Parisini, "Communication-efficient stochastic distributed learning," *IEEE Trans. Autom. Control*, early access, Mar. 18, 2026, doi: 10.1109/TAC.2026.3675513.

[20] N. Bastianello, R. Carli, L. Schenato, and M. Todescato, "Asynchronous distributed optimization over lossy networks via relaxed ADMM: Stability and linear convergence," *IEEE Trans. Autom. Control*, vol. 66, no. 6, pp. 2620–2635, Jun. 2021.

[21] P. Giselsson and S. Boyd, "Linear convergence and metric selection for douglas-rachford splitting and ADMM," *IEEE Trans. Autom. Control*, vol. 62, no. 2, pp. 532–544, Feb. 2017.

[22] J. Dattorro, *Convex Optimization & Euclidean Distance Geometry*. Palo Alto, CA, USA: Meboo Publishing, 2010. [Online]. Available: https://www.convexoptimization.com/TOOLS/0976401304_v2010.10.26.pdf

[23] X. Chang, F. Nie, Y. Yang, C. Zhang, and H. Huang, "Convex sparse PCA for unsupervised feature learning," *ACM Trans. Knowl. Discovery Data*, vol. 11, no. 1, pp. 1–16, Feb. 2017.

Open Access funding provided by 'Università degli Studi di Cagliari' within the CRUI CARE Agreement